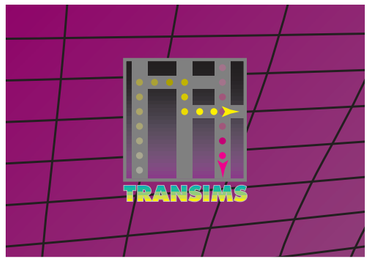
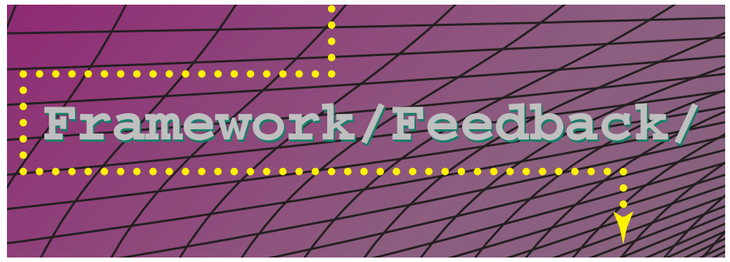
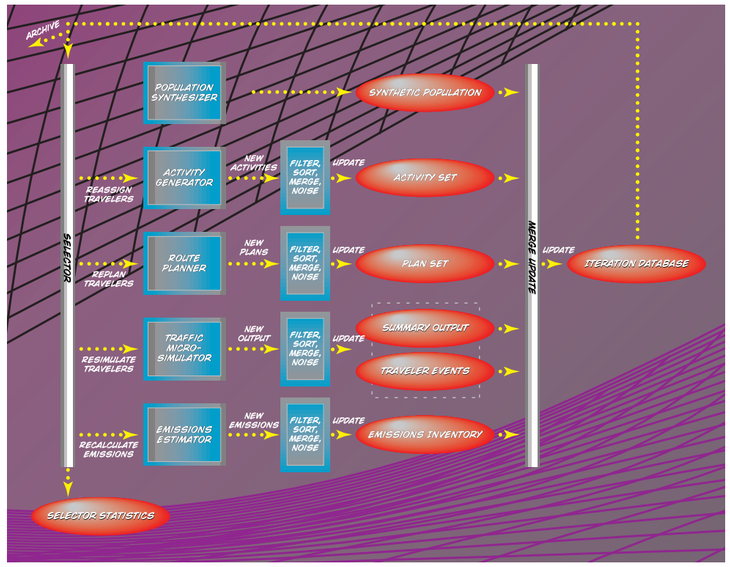
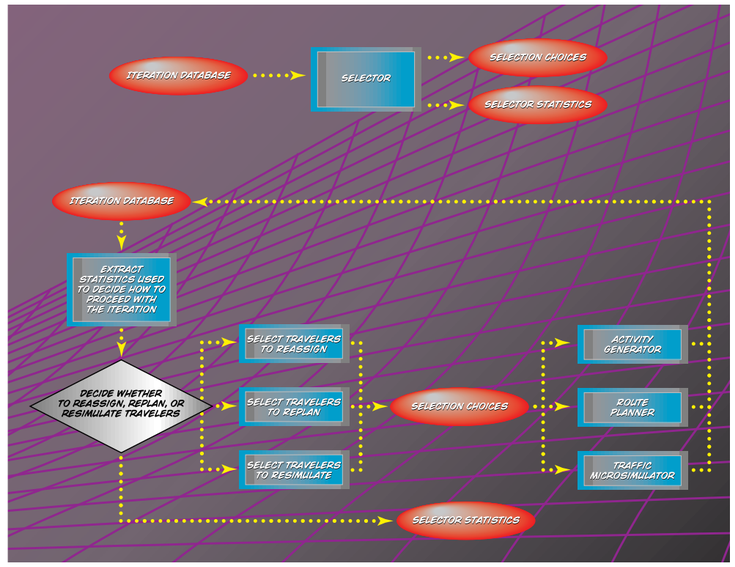




TRANSIMS FRAMEWORK

- Flexible software system
- For transportation planning studies/experiments
- Supports the future growth of TRANSIMS technology
- Building blocks
  - Software modules
    - Standardized command file
    - Standardized input/output interface requirements
    - Several major modules already available
    - Third-parties may replace or add new conforming modules
    - Reusable C++ libraries for building TRANSIMS objects (network, plan, activity, and simulation output)
    - High-performance, parallel/distributed computing
  - Simulation data files
    - Well-documented text formats
    - Interface library callable from C, C++, FORTRAN, etc.
  - Data manipulation tools
    - Filtering, sorting, indexing, merging, searching, summarizing, "noising"
    - For standard data files
  - Tools for controlling iteration between modules
    - "Iteration database" with history of iterations
    - "Selector" controlling and supervising iteration process
  - Iteration "scripts"
    - Define a study or experiment
    - Predefined for typical studies
      - Calibration
      - Sensitivity analysis
      - Convergence/equilibration of activities, plans, and traffic
- Many possible combinations of above "building blocks"
  - ⇒ Many possible realizations of TRANSIMS





SELECTOR: PURPOSE

- Controls when modules are run and how the data are routed between modules
- Operates in conjunction with an "iteration script" that provides the top-level control for a series of TRANSIMS iterations
- No single, "standard" selector component
  - Different study designs involve different iteration schemes
  - A variety of selectors have uses in different studies or other contexts

EXAMPLES OF SELECTION STRATEGIES

- **Replan routes** for travelers who have simulated travel times differing too much from their planned travel times
- **Reassign activities** for households if any member is too late for work
- **Average microsimulation output** from several runs

STRATEGY FOR ITERATIONS